基于课程间关联规则的排课优化算法研究 *

庞书杰，刘其成†，牟春晓

(烟台大学 计算机与控制工程学院，山东 烟台 264005)

摘 要：针对如今高校招生人数扩张所伴随而来的高校排课资源紧张的现象进行了研究，提出了一种基于关联规则的排课优化算法(SH-AP 算法)来优化高校排课。SH-AP 算法将关联规则算法应用到排课过程中，通过对高校学生选课数据进行关联规则的挖掘，找出需要排课的课程之间的关联规则，之后 SH-AP 算法再将已经挖掘出的课程间关联规则应用到排课系统中，对高校排课进行优化。实验最终得到课程间关联规则的数据，并为高校提供人性化的排课。根据研究与对比表明 SH-AP 算法在对排课的合理化与人性化上有很好的优化效果，帮助学校在进行排课的时候，考虑课程时间地点冲突问题的同时，对学生的需求进行满足。

关键词：学生选课；高校排课；关联规则

中图分类号：TP301.6　　doi: 10.3969/j.issn.1001-3695.2018.04.0249

Research on timetabling optimization algorithm based on inter curriculum association rules

Pang Shujie, Liu Qicheng†, Mu Chunxiao

(*School of Computer & Control Engineering, Yantai University, Yantai Shandong* 264005, *China*)

Abstract: For nowadays the enrolment expansion of colleges and universities to universities curriculum resources nervous phenomena are studied, proposed an optimization scheduling algorithm based on association rules (SH-AP algorithm) to optimize curriculum in colleges and universities. SH-AP algorithm applied the association rules algorithm in the curriculum arrangement process, through to the university students' course selection data for association rules mining, found out association rules between need arranging courses. SH-AP algorithm then had been digging out association rules between the course of curriculum arrangement system, applied to optimize the curriculum in colleges and universities. The experiment finally obtains the data of the association rules between the courses, and provides the humanized curriculum for the university. According to the research and comparison, the SH-AP algorithm has a good optimization effect on the rationalization and humanization of the scheduling. This algorithm can help the school to consider the conflict of time and place in the course of scheduling, and satisfy the needs of students.

Key words: students' elective courses; college scheduling; association rules

0　引言

如今的人们平均受教育水平提高，越来越多的人走进了大学校园，全国各个高校的招生人数大幅度提高。而高校一年比一年多的新生入学，导致如今高校教学资源极度紧张[1]。当代大学生在大学期间也已经不仅仅满足对自己本身专业知识的需求，都想要更加全面的发展自己。而每个学生在专业知识的基础上，想要扩展的方向也各不相同。因此，各高校在给大学生们安排了专业必修的专业课程外，还为广大大学生们提供了各种方面的包括文科、理科和工科的选修课程[2]。

高校选修课的设立，使得学生对自己的学习方向有了更高的选择性。而学生在选择课程的时候会留下大量数据，这些数据包含了学生喜欢的课程间的关系。同时，选修课的设立加大了学校的排课压力，如何在对所有课程的上课时间地点进行合理的排课的同时尽量满足学生的选课需求，使得学生喜欢的课程不会产生冲突，是现如今高校排课所迫切需要解决的问题[3]。

同时随着大数据被用于服务教育和科学一个共同利益目标，教育数据挖掘方面的技术也得到了迅猛发展。如关联规则、分类和聚类的数据挖掘技术在教育数据中得到了灵活运用。通过对高校选课数据的信息数据挖掘，可以发现其中的课程间的关联规则。高校管理者可以根据这些挖掘的关联规则，更好的来进行高校课程间的排课，避免出现课程排课冲突。

收稿日期：2018-04-12；**修回日期：**2018-05-18　　**基金项目：**山东省自然科学基金资助项目（ZR2016FM42）；山东省重点研发计划资助项目（2016GGX109004）；国家海洋局“十三五”海洋经济创新发展示范重点项目（YHC-ZB-P201701）；国家自然科学基金资助项目（61702439）

作者简介：庞书杰（1995-），男，河南新乡人，硕士研究生，主要研究方向为数据挖掘、并行运算；刘其成（1970-），男（通信作者），教授，博士，主要研究方向为云计算、大数据、多 agent 系统、数据挖掘等（ytliuqc@163.com）；牟春晓（1980-），男，实验师，硕士，主要研究方向为云计算、大数据等.

关联规则算法是数据挖掘算法中十分重要的组成部分，其中 Apriori 算法更是十大经典数据算法之一，也是影响最为深远的算法之一[4]。但它在挖掘关联规则时，会产生大量候选项集，以及大量的 I/O 的开销[5]。DHP 算法是 Apriori 的改进算法，它在 Apriori 之上，优化了对候选项集的筛选，使其不必再大量的访问数据，减少了 I/O 的开销。

本文提出的基于课程间关联规则的排课优化算法 SH-AP 通过对往届进行关联规则挖掘，从而尽最大可能的减少排课中出现的排课冲突问题。

1 相关工作

关联规则算法自被提出到现在一直是数据挖掘中十分重要的组成部分之一。各式各样的关联规则算法被提出，并在多方面获得了广泛的应用，如在金融业、销售业以及监控预测行业中的应用[6]。当前许多专家学者对关联规则算法相关的内容以及应用进行了大量研究。

文献[7]将关联规则算法那应用于气象观测设备中，来保证气象观测设备收集数据时的稳定性。文献[8]通过分析现有的图书推荐系统的不足之处，提出通过关联规则算法来对图书借阅数据进行挖掘，以便提供更人性化的图书推荐服务。文献[9]通过一种改进的关联规则算法对电子病历系统中的历史数据进行挖掘，找出电子病历系统中的有用数据，为医生诊诊断提供帮助。文献[10]则是根据关联规则算法对食品生产加工数据进行处理，对食品安全问题的原因进行诊断，并进一步进行预警。以上研究都是从不同的应用角度，通过使用关联规则算法进行数据挖掘。但伴随着关联规则的挖掘会产生大量的候选项集，需要不断的扫描数据库，造成极大的 I/O 开销。

本文是从高校排课的角度使用关联规则算法进行优化。高校选课排课也是高校和大学生的主要任务之一，也是促进学生发展的全面化的重要组成部分。当前已经有部分专家学者对高校的选课排课情况进行研究。文献[11]根据高校现如今的学分制的基础下，以课程为中心构建计算机排课系统的模型。对此模型上的排课问题进行讨论并给出了解决办法。文献[12]中针对学生与课程数量的增大，设计实现了教务排课管理系统来提高排课工作的准确性和及时性。文献[13]中通过关系型数据库系统，研究了一种微机的排课系统，促进了教育管理的信息化和现代化。以上是研究通过不同方面来对高校排课的准确性和合理性进行研究和应用。但文献[11]仅仅讨论了学分对于高校排课的影响，忽视了学生对于课程选择的主观性，并不全面。文献[12]与文献[13]中，则都是直接对学校的所有课程进行排课，仅仅考虑到了学校方面的教学安排，而忽视了学生的选课需求。

为了解决上述问题，本文提出了基于关联规则 DHP 算法的高校排课优化算法(SH-AP 算法)，其目的是为了通过对学生选课数据进行关联规则分析来实现对高校排课的进一步优化。在兼顾高校排课的同时根据学生选课喜好对排课进行优化。本文先将学生选课数据进行预处理，然后根据选课数据，通过 SH-AP 算法迅速筛检课程集合的候选项集，挖掘出表示课程间关系的阈值。最后根据课程间关系的阈值对排课进行优化，使排课能更加人性化与合理化。

2 SH-AP 算法设计

本文提出的 SH-AP 算法是基于 DHP 的高校排课优化算法。相较传统 Apriori 算法减少了候选项集的数量，减少 I/O 开销。将高校学生的选课数据通过该算法进行挖掘，为高校的排课提供优化。

2.1 DHP 算法简介

关联规则是由两个阈值确定，分别为最小支持度和最小置信度。其中最小支持度是指满足支持度的最小值，支持度则是一个项集在数据库中出现的频率。另一个阈值，最小置信度是指满足置信度的最小值，置信度则是通过支持度计算得到，表示一个项目集合中包含 A 的概率下，包含 B 的概率[14]。

关联规则算法是根据最小支持度得到频繁 1 项集，连接得到候选 2 项集。再根据最小支持度对各项的进行计数并对候选 2 项集剪枝，得到频繁 2 项集。然后继续循环向下去，直到不再产生新的候选项集或者不再产生新的频繁项集为止。得到频繁项集后在根据相应的置信度算法算出置信度，得到关联规则。

DHP 算法设计的思想初衷，就是为了针对传统关联规则算法 Apriori 算法在挖掘关联规则时所产生的很大规模的候选项集。DHP 算法的作用主要体现在 Apriori 算法中剪枝的过程中，主要是用来减少 k 项候选项集的个数。

DHP 算法主要分为四个部分：

1） 扫描数据库 D

扫描数据库 D，得到数据库 $D=\{T_1,T_2,T_3\cdots,T_n\}$，$T_i$ 表示每项事务集合，以及项目集 $I=\{I_1,I_2,I_3,I_4,I_5\}$。每一个事务 $T_i(i=1,2\cdots,n)$满足 $T_k\in I$，而所有的事务 T_i 都分别是项目集合 I 的一个子集。

2) 构建 Hash 表

扫描数据库 D。在第 k 次扫描时，每项事务产生自身的 k+1 项集，将生成的 k+1 项集带入到 Hash 函数当中得到哈希值。根据其哈希值放入到该哈希值对应的桶中，并对桶中元素个数进行计数。

3) 根据 Hash 桶筛选候选项集

对于桶中元素个数的计数，若桶中元素个数小于最小支持度，则该哈希值所对应桶中元素一定不是频繁项集， 将其从候选项集中删除。

4) 根据候选项集求得频繁项集.

根据得到的候选项集 C_{k+1} 扫描数据库，统计其支持度个数，

若大于最小支持度则加入到频繁项目集 L_{k+1}。得到 L_{k+1} 后继续返回道筛选下一个 C_{k+2} 候选项集以此循环，直到没有新的候选项集产生或者没有新的频繁项集产生。

2.2 SH-AP 算法设计

SH-AP 算法主要是分为三个部分。

a)首先将学生们的选课数据进行预处理，这部分是接下来进行关联规则挖掘的必要部分。将学生们的选课数据做为训练集，$D=\{T_1,T_2,T_3\ldots,T_n\}$作为学生选课的数据集的集合。其中 T_n 标志的每一个事务表示为学生 ID，之后每个事务包含的项目集为该学生选课的情况。 $I=\{I_1,I_2,I_3,I_4,\ldots I_n\}$则是表示可供学生选课的课程集合。课程包括了计算机导论、Java 面向对象和单片机等 11 项学科课程。

b)学生课程间关联规则的数据挖掘阶段。将学生数据训练集带入到关联规则算法中进行数据挖掘，得到各个课程之间的关联规则以及该关联规则的支持度和置信度。

c)对已经得到的关联规则以及它的置信度和支持度进行分析，增加一个新的阈值提升度。根据提升度的值对课程间的关系进行排序，将排序结果反馈给高校管理人员。高校管理人员根据得道的关联规则以及提升度的值对排课的安排进行优化处理。

SH-AP 算法的三个部分中，第一部分数据预处理阶段是由人工完成，将课程数据转换为数字字母代替。第二部分以及第三部分是 SH-AP 算法的核心部分。

2.2.1 学生课程的关联规则挖掘

扫描学生数据库 D 得到学生选课数据集，每个事务包含的项目集为该学生选课的情况。例如，项目集 $I=\{I_1,I_2,I_3,I_4,\ldots I_n\}$为学生选课的课程集合，学生在其中任选两项以上课程。算法的执行条件设为最小支持度满足 minsup 大于等于 0.2，即在选课的总数据库中该选课的课程集合元素出现次数至少为 20%。

根据学生的选课数据，将满足最小支持度的课程组合放入频繁项目集 L 中。

首先，扫描学生选课数据库生成选课课程的候选 1-项集 $C_1=\{\{I_1\}\{I_2\}\{I_3\}\{I_4\}\{I_5\}\}$。并统计 C_1 中每项课程的支持度，求出学生所选课程满足支持度的课程频繁 1-项集 L_1。将数据库中每个学生自己选择的课程进行连接组合，得到每个学生所选课程集合本身的课程候选 2-项集。 算法描述如下：

```
Input: student data
Output: c_k
for each Student Data do
(1)    Cdhp={ Courses for each student};
(2)    L1={ each course ∈ Cdhp};
(3)    L2={ each course ∈ Cdhp};
(4)    if ((L1[k-1]<L2[k-1])&&L1[n]=L2[n])
          /*n={1,2,3…k-2}*/;
(5)       c=L1connect L2;
(6)     HashFunction(c);
 end
```

将所得到的每个学生得到的候选项集 c_k 中的课程组合，分别通过哈希函数进行分类。通过 hash 函数计算出课程集合中每个课程组合的哈希值，即该课程组合的存储地址。将地址相同的课程集合中的课程组合放在同一集合桶中，并对该地址下的集合桶中的课程集合元素的个数进行计数。算法描述如下：

```
Input: Student data
Output: L_k
for each student
(1)     for c_k∈each student
(2)        {x,y}∈C_k;   /*course combination*/
 (3)        HashValue ( x , y) = ( ( order of x) ×10 + ( order of y ) )
            mod7;
(4)        HashB [HashValue][1] add 1;
(5)     for each L1×L1
(6)        {x,y}∈ L1×L1;
 (7)        HashValue ( x , y) = ( ( order of x) ×10 + ( order of y ) )
            mod7;
(8)        If(HashB [HashValue][1]>=minsup)
(9)        {x,y}add to C_k add;
(10)    for each c∈C_k
(11)       for each T∈D
(12)       if c∈T c.sup add 1;
(13)       if c.sup>=minsup add c to L_k;
 end
```

之后，根据课程频繁项目集 L_1 通过自连接得到一个 2-项集的课程组合集合，将其中元素带入到 Hash 函数中得到哈希值。将其哈希值对应桶的计数小于 minsup 的元素从候选项集中删除，其余加入候选项集。之后对候选项集计数，大于 minsup 的加入到频繁项集 L 中。之后循环下去得到所有频繁项集。

在传统的 Apriori 算法中，对学生数据挖掘的时候会产生大量的课程集合候选项集。在对候课程集合计数阶段，每个候选项集的计数都需要对学生数据库进行一次扫描，最终挖掘出课程间关联规则需要多次扫描数据库。例如 10^4 个频繁 1-项集就可能产生 10^7 个候选 2-项集。如此多的候选项集需要多次的扫描数据库，从而造成了大量的 I/O 开销。

本文提出的 SH-AP 算法通过减少候选项集的个数来减少需要扫描数据库的次数，从而减少 I/O 开销。在课程频繁项集自连接产生的课程集合后，通过对比之前根据每个学生选课得到的课程候选项集所建立的 Hash 桶集合来进行筛选，删除桶中计数不满足最小值支持度的课程集合，从而得到候选项集。

相比传统 Apriori 算法减少了产生的候选项集的个数。候选项集的减少使得在对每个课程集合的候选项集计数的时候扫描学生数据库的次数减少，从而减少了 I/O 开销。

2.2.2 基于关联规则的排课优化

通过第二部分关联规则的挖掘，得到了学生选课的课程之间的关联规则和它的支持度以及置信度。这里加入提升度 Lift 作为一个新的阈值。根据课程间的提升度的大小来确定高校排课时课程排课的优先级。提升度越大，关联性越高。

读取数据库 D_1 得到已初次排列的课表。D_1 为周一到周五的排课数据，其中 $X_i=\{T_1,T_2,\cdots,T_5\}$表示一天的课时，$T_n=\{i|i\in I\}$ 则是表示的在课时 T_n 时需要上课的课程，$C_i=\{(course,Lift)|course\in I\}$则代表了与所有课程 I_i 相关的课程和提升度 Lift 的集合。每个课程根据必修课和选修课不同都有一个标记 star，若 star 为 1 或 0 分别为必修课与选修课。

扫描数据库，根据课程关联度的大小对课表进行优化，得到优化过的课表 D_2。算法描述如下：

```
Input: Timetable data D1
Output: Schedule D2
for each Ix∈I    / *Ix∈Tn(n=1,2,..,5)*/
(1)      mix=0
(2)      for each Lift∈Cx
(3)          if(Lift<mix) mix=Lift;
(4)      for i<=Cx.size
(5)          if i.lift == mix    c= i.course;
(6)      if c∈Tn
(7)      for each T∈D//T is one period
(8)          if T != Tn
(9)          for each Ij∈T
(10)             if Ij.star <= c.star
(11)       note=c
(12)       for each Cj
                  /*prevent error replacement*/
(13)                 if(Cj.lift >= mix)
(14)                     if(Cj.course∈Tn ) continue;
(15)                     x=c; c=Ij; Ij= x;break;
(16)         if note == c break;
  end
```

上述程序根据提升度的大小，将高校的排课课表进行了进一步的优化，找到课程 C_x 关联度最高的课程 c，并从别的时间段中找到能够与之替换的课程，将两个课程替换，避免关联度高的课程冲突。

SH-AP 是对高校排课进行优化的算法，在原有排课的基础上增加了课程关系提升度这一阈值对已有的高校排课进行了进一步排课优化，提高了高校排课对于学生学习兴趣方向的包容性和精确性。

3　实验结果与分析

3.1　实验数据

本次实验所采用的数据是国内某高校的某学院的学生的大学本科期间的选课情况。其中包括了该专业学生大学四年的校任选课以及专业选修课的选课情况。实验数据挑选了该专业各个方向的选修课，保证了选课课程的全面性与覆盖面。

数据集一共是有 527 个事务集组成，每一个事务集都是每个学生大学本科期间的选课情况。项目集包括计算机导论、数据库原理、艺术鉴赏、影视鉴赏、Java 面向对象、Android 课程学习、Flash 动画制作与欣赏、图形学原理与应用、机器视觉、单片机原理以及嵌入式系统等。在数据预处理中，将各类课程通过不同的数字代替。

3.2　评价标准

关联规则的可信度以及实用性的评测指标为其支持度与置信度。在对高校选课排课的关联规则进行挖掘的时候，需要得到课程之间的关联规则的三个评价指标：支持度、置信度和提升度。

1)支持度（support）

通俗地讲，一组项目出现的频率就是其支持度，即项目在总的数据集 D 中出现的次数。定义上讲支持度就等于该项目在事务集 D 中出现的次数比上事务数据集 D 的总数的比值。

$$Support(X)=P(X)=\frac{|T:X\in T,T\in D|}{|D|}\times 100\% \quad (1)$$

其中：X 为所求支持度的项目，T 为项目 X 所在的事务 T，T 属于事务集 D。分子为包含项目 X 的事务 T 的个数，分母为事务集 D 的总个数。

2)置信度（confidence）

在挖掘关联规则的中，要产生关联规则就是要通过上面计算的支持度来进一步产生规则，而产生的规则则是要满足最小置信度的要求。置信度就是项目集合 C 中已经包含 X 的条件下包含 Y 的概率，即条件概率。

$$Conf(X\rightarrow Y)=\frac{Support(X\cup Y)}{Support(X)}\times 100\% \quad (2)$$

X 与 Y 的支持度比上 X 的支持度，就可以得到 X 与 Y 产生的关联规则的置信度。

3)提升度（lift）

提升度是指组合{A,B}中选择 A 的同时也选择了 B 的比例与选择 B 的所占总数的比例的比值。

$$Lift(A\rightarrow B)=\frac{Conf(A\rightarrow B)}{Support(B)}=\frac{P(A\&B)}{P(A)P(B)} \quad (3)$$

其中：lift 就是代表了组合的提升度。P(A&B)/P(A)就是组合{A,B}的置信度 conf，P(B)则是 B 的基于数据集的支持度 sup。如果产生的关联规则能够满足最小置信度 minsup 的阈值和最小置信度 minconf 的阈值，以及 Lift 的值大于 1 就可以认为该关联规则是可信的，有参考价值的。

3.3 结果分析

在学生选课关联规则的实验测试中事务集 D 为 527 个学生的选课数据的总集合，其中课程项目集合 I 有 11 门课程。将学生的选课数据进行处理，课程通过数字代替。如表 1 所示。

表 1 数据预处理

TID	课程
T100	4,9,10,11
T200	2,5,6,7
T300	1,3,4,6,7,10,11
T400	2,4,5,6,11
T500	2,5,6,8,11
T600	1,2,3,6,10,11
T700	1,3,8,11
T800	3,4,5
…	…

表 1 中展示了部分学生选课数据。将这些处理过的数据带入到高校排课优化算法 SA-HP 中，根据设置的最小支持度 minsup 的阈值以及最小置信度 minconf 的阈值进行关联规则的挖掘。

通过选课课程关联规则算法的挖掘得到了关联规则，以及各项关联规则的频率、支持度和置信度。表 2 中展示了挖掘结果。

表 2 实验结果

关联规则	频率	support	confidence	lift
2⇨5	192	0.36	0.80	1.33
5⇨6	201	0.38	0.78	1.59
8⇨2	116	0.22	0.80	1.78
8⇨5	106	0.20	0.73	1.22
{2,6}⇨5	134	0.25	0.93	1.55

实验结果可以看出，得到的关联规则的支持度大于 0.2，置信度大于 0.7 以及提升率大于 1。关联规则中 2、5、6 和 8 分别代表了课程：数据库原理与应用、Java 面向对象、Android 课程学习和图形学原理与应用。2 与 5、5 与 6、8 与 2、8 与 5 以及 2、6 与 5 产生了强关联规则，它们的课程同时受到了多名学生的选择。在高校排课优化时应优先对 Lift 值高的 8 与 2 课程进行排课，然后根据 Lift 大小依次排序避免这几门课程之间的时间以及空间上的冲突。

根据以上实验得到的结论，以及测试的指标达到的数值。可以看出基于关联规则的高校排课优化算法(SH-AP 算法)可以对高校排课起到很好的优化作用。

3.4 相关工作对比

SH-AP 算法与基于 Apriori 的高校排课算法，以及基于文献[15]中提出的 F2AM 算法的高校排课算法进行比较，其中 F2AM 算法是近几年提出的关联规则算法。实验环境为：CPU 为 Intel(R) Core™ i7-7700，主频 3.60 GHz，内存为 8 GB。算法都是采用 Java 语言进行编写，JDK 使用 1.8.0 版本，在 Eclipse 的环境下进行编译与运行。

图 1 表示了不同最小支持度下的 Apriori 算法、F2AM 算法与 SH-AP 算法运行时间的大小。不同的最小支持度下比较了三种算法在不同最小支持度下的算法的性能。由图可知，在最小支持度较低的时候，三种算法的时间花销

相差较大，F2AM 算法虽然优于 Apriori 算法，但 SH-AP 算法也明显优于 F2AM 算法。而伴随着最小支持度的提高，扫描数据库的迭代次数减少，产生的候选项集以及频繁项集的个数大量减少，三种算法的时间开销都比较低。但 SH-AP 算法的运行时间也都小于 Apriori 算法与 F2AM 算法。

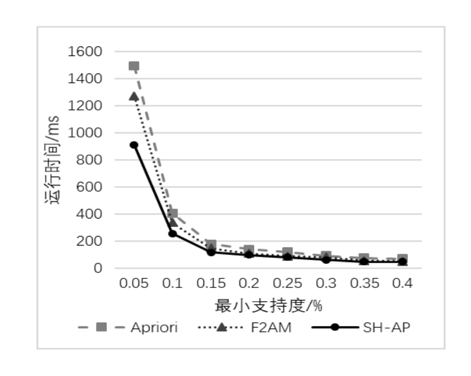


图 1 不同支持度的算法对比

图 2 则表示了在相同最小支持度，不同的最小置信度的条件下，Apriori 算法、F2AM 算法和 SH-AP 算法运行时间的大小的比较。比较了三种算法在最小支持度为定值，不同最小支持度下的算法的性能。由图可知在不同的置信度下，三种算法的运算时间波动不大，这主要是由于求得置信度部分，主要是基于支持度上的数学运算。之前求支持度时，已将各项集的计数存储，运算所需开销较低。由表可知，虽然 F2AM 算法在运算时间上优于 Apriori 算法，但本文提出的 SH-AP 算法的运行时间也同样明显优于以上两种算法。

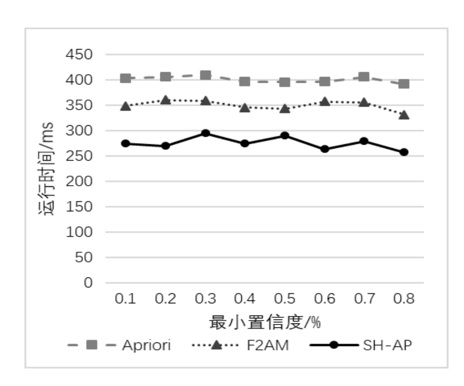


图 2 不同置信度的算法对比

通过比较三种算法不同支持度下的运行时间，以及对比三

种算法不同置信度下的运行时间，可以看出在对大量学生选课数据进行关联规则挖掘的时候，SH-AP 算法的挖掘效率明显高于 Apriori 算法与 F2AM 算法。这也说明了 SH-AP 算法在对学生选课课程间关联规则分析的方面有很好的效果。

4 结束语

本文提出了一种基于课程间关联规则的排课优化算法(SH-AP 算法)。根据学生们选课产生的数据，通过数据挖掘中的关联规则算法对课程间关系进行发掘，得到课程间关系的提升度阈值。将表示课程间关系的提升度带入到高校排课过程中，根据课程间的关系对高校排课进行优化，使得高校排课在避免课程的上课时间地点冲突的同时，兼顾学生选课的需求。使得高校排课更加的合理化与人性化。

如今，随着高校大量的扩招学生，以及高校更多的新型课程的设立，每年都会产生大量的学生选课数据。数据量的增大，现有的串行运算方式将不再适合对大量的学生选课数据进行处理，影响工作效率。下面要根据数据大量提升的情况，将关联规则算法并行化，方便处理大量数据，减少挖掘课程关系是所耗的时间。

参考文献：

[1] 姚双良. 数据挖掘在高校课程相关性中的应用研究 [J]. 科技通报, 2012, 28 (12): 232-234. (Yao Shuangliang. Application and research on correlation between colleges courses based on data mining [J]. Bulletin of Science and Technology, 2012, 28 (12): 232-234.)

[2] 姜永超. 基于数据挖掘的学生选课及学习行为分析算法研究 [J]. 现代电子技术, 2016, 39 (13): 145-148. (Jiang Yongchao. Research on students course selection and learning behavior analysis algorithm based on data mining [J]. Modern Electronics Technique, 2016, 39 (13): 145-148.)

[3] 王念桥, 姚四改. 基于改进粒子群优化算法的排课问题 [J]. 计算机应用, 2013, 33 (1): 207-210. (Wang Nianqiao, Yao Sigai. Curriculum scheduling based on improved swarm optimization algorithm [J]. Journal of Computer Applications, 2013, 33 (1): 207-210.)

[4] 杜孝平, 马秀莉, 唐世渭, 等. 快速关联规则挖掘算法 [J]. 计算机工程与应用, 2002, 38 (11): 1-4. (Du Xiaoping, Ma Xiuli, Tang Shiwei, *et al.* A fast algorithm for mining of association rules [J]. Computer Engineering and Applications, 2002, 38 (11): 1-4.)

[5] 闫梦洁, 罗军, 刘建英, 等. IABS: 一个基于 Spark 的 Apriori 改进算法 [J]. 计算机应用研究, 2017, 34 (8): 2274-2277. (Yan Mengjie, Luo Jun, Liu Jianying, *et al.* IABS: Parallel improved apriori based on spark [J]. Application Research of Computers, 2017, 34 (8): 2274-2277.)

[6] 崔贯勋, 李梁, 王柯柯, 等. 关联规则挖掘中 Apriori 算法的研究与改进 [J]. 计算机应用, 2010, 30 (11): 2952-2955. (Cui Guanxun, Li Liang, Wang Keke, *et al.* Research and improvement on apriori algorithm of association rule mining [J]. Journal of Computer Applications, 2010, 30 (11): 2952-2955.)

[7] 李涛, 郁美辰, 陆正邦, 等. 基于关联规则挖掘的气象观测设备一致性检测算法 [J]. 电子测量与仪器学报, 2017, 31 (10): 1568-1573. (Li Tao, Yu Meichen, Lu Zhengbang, *et al.* Consistency detection algorithm for meteorological observation equipment based on association rules mining [J]. Journal of Electronic Measurement and Instrumentation, 2017, 31 (10): 1568-1573.)

[8] 陈定权, 朱维凤. 关联规则与图书馆书目推荐 [J]. 情报理论与实践, 2009, 32 (6): 81-84. (Chen Dingquan, Zhu Weifeng. Association rules and library bibliographic recommendation [J]. Information Studies: Theory & Application, 2009, 32 (6): 81-84.)

[9] 丁卫平, 祁恒, 董建成, 等. 基于关联规则的电子病历挖掘算法研究与应用 [J]. 微电子学与计算机, 2007, 24 (3): 69-73. (Ding Weiping, Qi Heng, Dong Jiancheng, *et al.* Research and application of the algorithms of the quick and effective date mining based on the association rules [J]. Microelectronics & Computer, 2007, 24 (3): 69-73.)

[10] 顾小林, 张大为, 张可, 等. 基于关联规则挖掘的食品安全信息预警模型 [J]. 软科学, 2011, 25 (11): 136-141. (Gu Xiaolin, Zhang Dawei, Zhang Ke, *et al.* The information Pre-warning model of food safety based on association rules mining [J]. Soft Science, 2011, 25 (11): 136-141.)

[11] 高怀雁, 梁志宏, 孙兴平. 学分制下以课程为中心的计算机排课系统设计 [J]. 计算机工程与应用, 2005, 41 (31): 212-214. (Gao Huaiyan, Liang Zhihong, Sun Xingping. The designing of computer courses arrangement system centralized on courses under the credit system [J]. Computer Engineering and Applications, 2005, 41 (31): 212-214.)

[12] 张建安, 杨学俊, 吴文一. 基于关联规则算法的排课系统设计与实现 [J]. 现代电子技术, 2010, 33 (2): 60-64. (Zhang Jianan, Yang Xuejun, Wu Wenyi. Design and implementation of online course arrangement system based on association rule algorithm [J]. Modern Electronics Technique, 2010, 33 (2): 60-64.)

[13] 修磊, 杨社堂. 高校排课管理系统的设计与实现 [J]. 太原理工大学学报, 2006 (S1): 75-77. (Xiu Lei, Yang Shetang. The design and implementation of the management system of the course arrangement in colleges and universities [J]. Journal of Taiyuan University of Technology, 2006 (S1): 75-77.)

[14] 崔妍, 包志强. 关联规则挖掘综述 [J]. 计算机应用研究, 2016, 33 (02): 330-334. (Cui Yan, Bao Zhiqiang. Survey of association rule mining [J]. Application Research of Computers, 2016, 33 (02): 330-334.)

[15] 纪怀猛. 基于频繁 2 项集支持矩阵的 Apriori 改进算法 [J]. 计算机工程, 2013, 39 (11): 183-186. (Ji Huaimeng. Improved apriori algorithm based on frequency 2-item set support matrix [J]. Computer Engineering, 2013, 39 (11): 183-186.)